



03-2: Spring Rest-Request Data Binding-Exception- OpenAPI

Trần Văn Thịnh



Spring Rest- Request Data Binding- Exception- OpenAPI

Request data binding

Validate Request Data

OpenAPI

Exception



Request Data Binding

@RequestParam

@PathVariable

@RequestBody

@RequestParam

- Lấy dữ liệu (query parameter) từ query String trong request url. Ví dụ <http://localhost:8080/foods?id=abc&type=mexico> thì query String là phần đặt sau dấu ?, được ghi dưới dạng:key=value.

```
//http://localhost:8080/foods?id=abc&type=mexico
@GetMapping("/foods") //Nếu đặt trùng @GetMapping("/food") sẽ báo lỗi khi biên dịch
public String getFoodByIdAndType(@RequestParam(name = "id") String foodId, @RequestParam String type) {
    return "Food ID: " + foodId + ", type: " + type;
}
```

- Mặc định trường nào dùng với **@RequestParam** phải có trong query param, nếu không sẽ báo lỗi. Nếu trường đó là không bắt buộc, thì khai báo thêm required=false

```
//http://localhost:8080/foods-plus?id=abc&type=mexico
@GetMapping("/foods-plus")
public String getFoodByIdAndTypeAndDescription(@RequestParam(name = "id") String foodId,
    @RequestParam String type, @RequestParam(required = false) String description) {
    return "Food ID: " + foodId + ", type: " + type + ", description: " + (description == null ? "No description" : description);
}
```

@RequestParam

- Nếu muốn lấy tất cả các param trong query string, thì sử dụng @RequestParam Map<String,String> allParams

```
//http://localhost:8080//api/foos?id=abc&type=mexico
@GetMapping("/api/foos")
public String getFoos(@RequestParam Map<String,String> allParams) {
    return "Parameters are " + allParams.entrySet();
}
```

- Với @RequestParam thì parameter được sử dụng URL decoded. Ví dụ: "id=abc+d" thì khi xử lý lấy ra RequestParam id sẽ có giá trị là "abc d"

@PathVariable

- Lấy dữ liệu (path variable) từ uri path (đường dẫn)

```
//http://localhost:8080/city/id/1
@GetMapping("/city/id/{cityId}")
public String getCityID(@PathVariable int cityId) {
    return "City id: " + cityId;
}
```

- Mặc định trường nào dùng với **@PathVariable** phải có trong uri path, nếu không sẽ báo lỗi. Nếu trường đó là không bắt buộc, thì khai báo thêm required=false

```
//http://localhost:8080/city/vietnam/hanoi
@GetMapping("/city/{countryName}/{cityName}")
public String getCity(@PathVariable(required=false) String countryName, @PathVariable String cityName) {
    return "City name: " + cityName + (countryName == null ? "" : " of country: " + countryName);
}
```

- Giá trị sử dụng trong **@PathVariable** là giá trị chính xác, không decode như **@RequestParam**

@RequestBody

- Map HttpRequest body với 1 object, có "automatic deserialization" để chuyển thành java object. Mặc định sử dụng định dạng json

```
0 //http://localhost:8080//api/foods/ (Method=POST) (Use postman to run)
1 @PostMapping("/api/foods")
2 public ResponseEntity<Food> createUser(@RequestBody Food food) {
3     return ResponseEntity.ok(new Food(10, food.getName(), food.getType(), food.getDescription()));
4 }
5 }
```



Demo:

<https://github.com/TechMaster/SpringBootBasic/tree/main/form/pathquery>

Validate Request Data

Validate Request Data

- Mặc định với Spring MVC xử lý sẵn các trường hợp như thiếu request param, path variable, type không đúng (type miss matched)...
- Trong trường hợp cần validate logic riêng, đơn giản, như lớn hơn bao nhiêu, nhỏ hơn bao nhiêu... thì có thể sử dụng các annotation trong javax validation (**Java Bean Validation**)
- 1 số annotation hay dùng trong **JSR(JSR-303:BeanValidation1, JSR-380 BeanValidation2)**
 - **@NotNull, @NotEmpty, @NotBlank**
 - **@AssertTrue**
 - **@Size**
 - **@Min, @Max**
 - **@Email**
 - **@Positive, @PositiveOrZero, @Negative, @NegativeOrZero**
 - **@Past, @PastOrPresent, @Future, @FutureOrPresent**

Validate Request Data

- Dependencies để sử dụng Validation API (not spring-boot)

- Validation API

```
<dependency>  
  <groupId>javax.validation</groupId>  
  <artifactId>validation-api</artifactId>  
  <version>2.0.1.Final</version>  
</dependency>
```

- Validation API Reference Implementation

```
<dependency>  
  <groupId>org.hibernate.validator</groupId>  
  <artifactId>hibernate-validator</artifactId>  
  <version>6.0.13.Final</version>  
</dependency>
```

- Cho spring boot, sử dụng spring-boot-starter-validation

Validate Request Data

- Với Spring boot, để sử dụng validator, cần sử dụng thêm
 - @Validated (annotation thường sử dụng cho class level hoặc method group)
 - Hoặc @Valid (thường sử dụng cho field(parameter, Object))
 - Với @RequestParam và @PathVariable có thể sử dụng cả @Validated và @Valid, Với @RequestBody và dùng các annotation để validate trong POJO, thường dùng @Valid
 - Ngoài ra có thể viết các Custom validation annotation hoặc viết service code để validate

Validate Request Data

- Với Spring boot, để sử dụng validator, cần sử dụng thêm
 - @Validated (annotation thường sử dụng cho class level hoặc method group)
 - Hoặc @Valid (thường sử dụng cho field(parameter, Object))
 - Với @RequestParam và @PathVariable có thể sử dụng cả @Validated và @Valid, Với @RequestBody và dùng các annotation để validate trong POJO, thường dùng @Valid
 - Ngoài ra có thể viết các Custom validation annotation hoặc viết service code để validate

Validate Request Data

- Demo: <https://github.com/TechMaster/SpringBootBasic/tree/main/form/pathquery>

OPEN API

OpenAPI

- Swagger hay OpenAP là một opensource được sử dụng để phát triển thiết kế và làm các document cho các REST API và còn có một số tính năng khác
- Hỗ trợ các annotation giúp cho ứng dụng spring của ta có thể mô tả được các restapi, và đồng thời build một giao diện giúp người dùng có thể tương tác trực tiếp với các rest api đó.

OpenAPI 3.0

- <https://github.com/OAI/OpenAPI-Specification/blob/main/versions/3.0.0.md>

Swagger2	OpenApi 3.0	Description
@Api	@Tag	Đánh dấu 1 class là nơi chứa các API
@ApiModel	không còn	Đánh dấu 1 class là Swagger Model
@ApiModelProperty	@Schema	Bổ sung các thông tin cho data model
@ApiOperation	@Operation	Mô tả cho một API và response của nó
@ApiParam	@Parameter	Mô tả các parameter
@ApiResponse	@ApiResponse	Mô tả status code của response
@ApiResponses	@ApiResponses	Mô tả danh sách các status code của response

OpenAPI

- Demo

<https://github.com/TechMaster/SpringBootBasic/tree/main/rest/02OpenAPI/bookstore>